



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/074,092	02/12/2002	Darpan Dinker	5681-05700	6704

7590 03/21/2006
Robert C. Kowert
Conley, Rose, & Tayon, P.C.
P.O. Box 398
Austin, TX 78767

EXAMINER

DUONG, OANH L

ART UNIT	PAPER NUMBER
----------	--------------

2155

DATE MAILED: 03/21/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 10/074,092	Applicant(s) DINKER ET AL.	
	Examiner Oanh Duong	Art Unit 2155	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 03 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 12/27/2005.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-33 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-33 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 12/27/2005 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. Claims 1-33 are presented for examination.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

2. Claims 1-9, 13-24 and 28-33 are rejected under 35 U.S.C. 103(a) as being unpatentable over Applicant Admitted Prior Art (AAPA) in view of Jin et al. (Jin) (US 6,330,689 B1).

Regarding claim 1, AAPA teaches a plurality of nodes coupled together, wherein each one of the plurality of nodes is coupled to at least one other one of the plurality of nodes for communicating data between the nodes (Figs. 1 and 2), and

in-process node (Fig. 2) comprising:

an in-process client (client 201, Fig. 2); and

a distributed data manager (Distributed Data Manager 211, Fig. 2),

wherein the in-process client and the distributed data manager for the in-process node are configured to execute within the same computer process on the in-process node, and wherein the distributed data manager for the in-process node is configured to

communicate data with the in-process client in a non-serialized format and communicate data with other ones of the plurality of nodes in a serialized format (page 4 lines 12-18); and out-of-process node (Fig. 1) comprising an out-of-process client, wherein the out-of-process client is configured to execute within a different process than any distributed data manager, and wherein the out-of-process client is configured to communicate data with other processes or other ones of the nodes in a serialized format (page 2 line 30-page 3 line 3).

AAPA does not explicitly teach wherein the plurality of nodes in the system comprises at least one in-process node and at least one process node.

Jin teaches a server architecture wherein application can be run either in-process or out-of-process with the server program (see abstract). Jin teaches a data system comprising wherein the plurality of nodes in the system comprises at least one in-process node and at least one process node (col. 6 lines 42-48).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify AAPA to include both in-process and out-of process nodes as in Jin. One would be motivated to do so to offer the flexibility to run either or both in-process and out-of-process applications (Jin, col. 6 lines 56-57).

Regarding claim 2, AAPA teaches the system as recited in claim 1, wherein the in-process client is configured to request data from the distributed data manager for the

Art Unit: 2155

in-process node, and wherein, in response to the client's request, the distributed data manager for the in-process node is configured to return a reference to an object for the data without serializing the data (page 4 lines 15-18).

Regarding claim 3, AAPA teaches the system as recited in claim 2, wherein the distributed data manager for the in-process node is configured to receive a request for data from another node, serialize the requested data and send the serialized data to the requesting node (page 4 lines 5-15).

Regarding claim 4, AAPA teaches the system as recited in claim 1, wherein the distributed data manager for the in-process node is configured to receive serialized data from another node, de-serialize the data and store the de-serialized data as an object (page 4 lines 5-15).

Regarding claim 5, AAPA teaches the system as recited in claim 1, wherein said in-process client is configured to send an object reference for the data to the distributed data manager for the in-process node to store data in the in-process node (page 3 lines 30-31).

Regarding claim 6, AAPA teaches the system as recited in claim 1, wherein all data store operations performed by the distributed data manager in the in-process node

store data in a non-serialized object format in a data store of the in-process node (page 4 lines 5-18).

Regarding claim 7, AAPA teaches the system as recited in claim 1, further comprising a distributed data manager for the out-of-process node, wherein the distributed data manager for the out-of-process node is configured to communicate data with other processes or other ones of the plurality of nodes in a serialized format (page 2 line 30-page 3 line 3).

Regarding claim 8, AAPA teaches the system as recited in claim 7, wherein the out-of-process client is configured to execute in a separate process and communicate data with the distributed data manager for the out-of-process node in a serialized format (Fig. 1, page 1 line 30-page 2 line 2).

Regarding claim 9, AAPA teaches the system as recited in claim 7, wherein the out-of-process client is configured to request data from the distributed data manager for the out-of-process node, and wherein, in response to the client's request, the distributed data manager for the out-of-process node is configured to send the requested data to the out-of-process client in a serialized format (page 2 line 30-page 3 line 3).

Regarding claim 13, AAPA teaches the system as recited in claim 7, wherein the distributed data manager for the out-of-process node is configured to replicate data to one or more other ones of the plurality of nodes (page 2 lines 16-17).

Regarding claim 14, AAPA teaches the system as recited in claim 7

AAPA does not explicitly teach the distributed data manager for the out-of-process node is comprised within an application server, and wherein the out-of-process client is a web server coupled to the application server.

Jin teaches the distributed data manager for the out-of-process node is comprised within an application server, and wherein the out-of-process client is a web server coupled to the application server (col. 6 lines 30-35).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify AAPA to include the distributed data manager for the out-of-process node is comprised within an application server, and wherein the out-of-process client is a web server coupled to the application server as in Jin. One would be motivated to do so to accommodate dynamic content sessions in which the server dynamically generates and server a response that is tailored to client (Jin, col. 5 lines 62-65).

Regarding claim 15, AAPA teaches the system as recited in claim 1, wherein the distributed data manager for the in-process node is configured to replicate data stored in the in-process node to one or more other ones of the plurality of nodes (page 3 line

Art Unit: 2155

30-page 4 line 2).

Regarding claim 16, AAPA teaches the system as recited in claim 7

AAPA does not explicitly teach the distributed data manager for the in-process node is comprised within an application server, and wherein the in-process client is a web server coupled to the application server.

Jin teaches the distributed data manager for the in-process node is comprised within an application server, and wherein the in-process client is a web server coupled to the application server (col. 6 lines 30-35).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify AAPA to the distributed data manager for the in-process node is comprised within an application server, and wherein the in-process client is a web server coupled to the application server as in Jin. One would be motivated to do so to accommodate dynamic content sessions in which the server dynamically generates and server a response that is tailored to client (Jin, col. 5 lines 62-65).

Regarding claim 17, AAPA teaches a method, comprising:

an in-process client requesting data from a distributed data manager for an in-process node of a distributed data system (page 4 lines 4-5), wherein the in-process client and the distributed data manager for the in-process node execute within the same process on the in-process node (Fig. 2, page 3 lines 15-16);

if the requested data is present in a data store managed by the distributed data

manager for the in-process node:

the distributed data manager for the in-process node returning the requested data to the in-process client as an object without serializing the data (page 4 lines 3 and lines 15-18);

if the requested data is not present in the data store managed by the distributed data manager for the in-process node:

the distributed data manager for the in-process node retrieving the requested data in a serialized format from another node of the distributed data system (page 4 lines 3-8 and lines 14-15);

the distributed data manager for the in-process node de-serializing the data retrieved from another node into an object (page 4 lines 12-15); and

the distributed data manager for the in-process node returning the requested data to the in-process client as the de-serialized object (page 4 lines 8-10);

an out-of-process client requesting data from a node (Fig.2, page 2 lines 20-21); and

the out-of-process client receiving the requested data in a serialized format (page 2 line 30-page 3 line 3).

AAPA does not explicitly teach in-process node and out-of-process node are in the distributed data system.

Jin teaches a server architecture wherein application can be run either in-process or out-of-process with the server program (see abstract). Jin teaches in-process node and out-of-process node are in the distributed data system (col. 6 lines 42-48).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify AAPA to include both in-process and out-of process nodes as in Jin. One would be motivated to do so to offer the flexibility to run either or both in-process and out-of-process applications (Jin, col. 6 lines 56-57).

Regarding claim 18, AAPA teaches the method as recited in claim 17, further comprising: the distributed data manager for the in-process node receiving a request for data from another node; the distributed data manager for the in-process node serializing the requested data; and the distributed data manager for the in-process node sending the serialized data to the requesting node (page 4 lines 2-15).

Claim 19 does not teach or define any new limitation above claim 4 and therefore is rejected for similar reason.

Claim 20 does not teach or define any new limitation above claim 5 and therefore is rejected for similar reason.

Claim 21 does not teach or define any new limitation above claim 6 and therefore is rejected for similar reason.

Claim 22 does not teach or define any new limitation above claim 7 and therefore is rejected for similar reason.

Claim 23 does not teach or define any new limitation above claim 8 and therefore is rejected for similar reason.

Claim 24 does not teach or define any new limitation above claim 9 and therefore is rejected for similar reason.

Claim 28 does not teach or define any new limitation above claim 13 and therefore is rejected for similar reason.

Claim 29 does not teach or define any new limitation above claim 14 and therefore is rejected for similar reason.

Claim 30 does not teach or define any new limitation above claim 15 and therefore is rejected for similar reason.

Claim 31 does not teach or define any new limitation above claim 16 and therefore is rejected for similar reason.

Regarding claim 32, AAPA teaches a method, comprising:

an out-of-process client requesting data from a distributed data manager for an out-of-process node of a distributed data system (page 2 lines 20-21), wherein the out-of-process client and the distributed data manager for the out-of-process node execute in two distinct processes (page 1 lines 30-31);

if the requested data is present in a data store managed by the distributed data manager for the out-of-process node:

the distributed data manager for the out-of-process node returning the requested data to the out-of-process client as a serialized object (page 2 lines 20-23 and page 3 lines 1-3);

if the requested data is not present in the data store managed by the distributed data manager for the out-of-process node:

the distributed data manager for the out-of-process node retrieving the requested data in a serialized format from another node of the distributed data system (page 2 lines 23-24 and page 3 lines 1-3); and

the distributed data manager for the out-of-process node returning the requested data in a serialized format to the out-of-process client (page 2 lines 27-28);
an in-process client requesting data from a distributed data manager for an in-

process node of the distributed data system (page 4 lines 4-5), wherein the in-process client and the distributed data manager for the in-process node execute within the same process on the in-process node (Fig. 2, page 3 lines 15-16); and the in-process client receiving the requested data in de-serialized format page 4 lines 14-18).

AAPA does not explicitly teach in-process node and out-of-process node are in the distributed data system.

Jin teaches a server architecture wherein application can be run either in-process or out-of-process with the server program (see abstract). Jin teaches in-process node and out-of-process node are in the distributed data system (col. 6 lines 42-48).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify AAPA to include both in-process and out-of process nodes as in Jin. One would be motivated to do so to offer the flexibility to run either or both in-process and out-of-process applications (Jin, col. 6 lines 56-57).

Claim 33 does not teach or define any new limitation above claim 10 and therefore is rejected for similar reason.

3. Claims 10-12 and 25-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Admitted Prior Art (AAPA) in view of Jin, and further in view of Crites et al. (Crites) (US 6,097,380).

Regarding claim 10, AAPA teaches the system as recited in claim 7, wherein said out-of-process client is configured to send serialized data to the distributed data manager for the out-of-process node to store data to the distributed data manager for the out-of-process node (page 2 lines 14-16 and page 2 line 30-page 3 line3),

AAPA-Jin does not explicitly teach the distributed data manager for the out-of-process node is configured to store the data in its serialized format.

Crites teaches a network system wherein one or more servers and a plurality of available media streams is included (see abstract). Crites teaches the distributed data manager for the out-of-process node is configured to store the data in its serialized format (col. 2 lines 36-42 and col. 5 line19-20).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify AAPA-Jin to store the data in its serialized format as in Crites. One would be motivated to do so to enable application programs to render continuous media streams of different types and from different sources without being aware of the details of the media streams (Crites, col. 1lines 40-43).

Claim 11 does not teach or define any new limitation above claim 10 and therefore is rejected for similar reasons.

Claim 12 does not teach or define any new limitation above claim 10 and therefore is rejected for similar reasons.

Claim 25 does not teach or define any new limitation above claim 10 and therefore is rejected for similar reasons.

Claim 26 does not teach or define any new limitation above claim 10 and therefore is rejected for similar reasons.

Claim 27 does not teach or define any new limitation above claim 10 and therefore is rejected for similar reasons.

Response to Arguments

4. Applicant's arguments filed 12/27/2005 have been fully considered but they are not persuasive.

In the remarks, applicants argued in substances that

(A) Prior art teaches away from the claims.

As to point (A), a prior art reference must be considered in its entirety, i.e., as a whole, including portions that would lead away from the claimed invention. *W.L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540, 220 USPQ 303 (Fed. Cir. 1983), cert. denied, 469 U.S. 851 (1984), MPEP 2141.02 (VI).

(B) Jin does not teach or suggest “a client executing within the same process as a distribute data manager”. Jin is also silent with respect to the use of “non-serialized format” and “serialized format” for communication with different entities by a distributed data manager of an in-process node.

As to point (B), AAPA does teach a client executing within the same process as a distribute data manager (i.e., *each of the distributed data manager 211 shares a process space with one of clients 201*, Fig. 2 page 3 paragraph [0009]); distributed data manager for the in-process node is configured to communicate data with the in-process client in a non-serialized format (i.e., *in the in-process configuration data may be communicated between a distributed data manager and a client sharing the same process space, without the additional computation for serialization, or in non-serialized format*, page 4 paragraph [0012]); and communicate data with other one of the plurality of nodes in a serialized format (i.e., *to transmit data across process boundaries, data is serialized before transmission, transmitted and received in its serialized format, and de-serialized at the receiving end*, page 3 lines 1-3, and page 4 lines 12-15).

In addition, in response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

(C) There is no teaching or suggestion to combine the references

As to point (C), In response to applicant's argument that there is no suggestion to combine the references, the examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988) and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). In this case, **AAPA** teaches in-process node(s) (Fig. 2), and out-of-process node(s) (Fig. 1). **Jin** teaches a system includes both in-process node and out-of-process node (see col. 6 lines 42-48). It would have been obvious to one of ordinary skill in the art at the time of the invention was made to incorporate the in-process node and out-of-process node of **AAPA** in the architecture of **Jin**. One would be motivated to do so to advantageously offer the developers the flexibility to run either or both in-process and out-of-process applications (**Jin**, col. 6 lines 55-57), thereby providing higher performance at the risk of crashing the system and affording high reliability (**Jin**, col. 4 lines 55-62).

(D) Prior Art does not teach all data store operations performed by the distributed data manager in the in-process node store data in a non-serialized object format in a data store of the in-process node.

As to point (D), **AAPA** teaches all data store operations performed by the distributed data manager in the in-process node store data in a non-serialized object format in a data store of the in-process node (**AAPA** teaches in the in-process

Art Unit: 2155

configuration data may be communicated between a distributed data manager and a client *without* requirement for *serialization/deserialization*, page 4 lines 15-18, and distributed data manager 211A stores data in its data store 21A and return the data to the clients, page 4 lines 8-10. One of ordinary skill in the art will readily recognize that data is stored in a non-serialized format in a data store since data is returned to client from data store without *serialization/deserialization*).

(E) Prior art does not teach the out-of-process client sending serialized data to the distributed data manager at an out-of-process node of a distributed data system, and the data is stored in serialized format.

As to point (E), AAPA teaches the out-of-process client sending serialized data to the distributed data manager at an out-of-process node of a distributed data system (i.e., in the out-of-process, data crosses process boundaries when transmitted between distributed data manager and client. To transmit data across process boundaries, data is serialized before transmission, transmitted and received in its serialized format, pages 2-3 paragraph [0007]); Crites teaches storing data in serialized format (Examiner has given a broadest reasonable interpretation of "data in serialized format" as "data stream" in view of the specification of the invention (page 3 lines 7-8). Crites teaches mass storage devices 18 are stored a plurality of data streams, col. 2 lines 36-38. Therefore, storing data stream, disclosed by Crites, reads on storing data in serialized format of the claimed invention)

(F) Prior art does not teach the distributed data manager for the out-of-process node is configured to receive serialized data from another node.

As to point (F), AAPA does teach the distributed data manager for the out-of-process node is configured to receive serialized data from another node (i.e., a distributed data manager node may communicate with other distributed data manager nodes...data is serialized before transmission, transmitted and received in its serialized format, Fig. 1 pages 2-3 paragraphs [0005]-0007])

As a result, the cited prior art does disclose a system and method as broadly claimed by the applicants. Applicants clearly have still failed to identify specific claim limitations that would define a clearly patentable distinction over prior arts.

5. THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

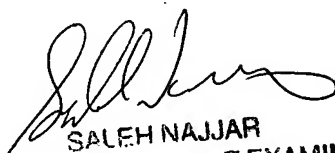
A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the mailing date of this final action.

6. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Oanh Duong whose telephone number is (571) 272-3983. The examiner can normally be reached on Monday- Friday, 9:30PM - 6:00PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Saleh Najjar can be reached on (571) 272-4006. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

O.D
March 13, 2006


SALEH NAJJAR
SUPERVISOR, PATENT EXAMINER